

103.1002.12

IN THE CLAIMS:

Please amend the claims as follows:

1. (Cancelled)

2. (Cancelled)

<sup>1</sup>

3. (Currently Amended) A method for recording a plurality of data about a plurality of blocks of data stored in a storage system, comprising the step of steps of: maintaining multiple usage bits for each of said plurality of blocks; ~~blocks;~~ wherein one bit of said multiple usage bits for each of said plurality of blocks indicates a block's membership in an active file system and plural bits of said multiple usage bits for each of said plurality of blocks indicate membership in plural read-only copies of a file system; and

<sup>DI</sup> wherein storing, in said storage system, said multiple usage bits for each of said plurality of blocks are maintained in said active file system.

<sup>2</sup>

4. (Previously Presented) A method as in claim <sup>1</sup>3, wherein one or more bits of said multiple usage bits for each of said plurality of blocks further indicate block reusability.

103.1002.12

3

3. (Currently Amended) A method for generating a consistency point for a storage system, comprising the steps of:

marking a plurality of inodes pointing to a plurality of modified blocks in a file system stored on said storage system as being in a consistency point;

flushing regular files to said storage system;

flushing special files to said storage system;

flushing at least one block of file system information to said storage system; and

queuing dirty inodes after said step of marking and before said step of flushing at least one block of file system information; ~~and requeuing any of said dirty inodes that were not part of said consistency point after said step of flushing at least one block of file system information~~

wherein said step of flushing said special files to said storage system further comprises the steps of:

pre-flushing an inode for a blockmap file to an inode file;

allocating space on said storage system for all dirty blocks in said inode and said blockmap files;

flushing said inode for said blockmap file again;

updating a plurality of entries in said blockmap file wherein each entry of said plurality of entries represents a block in said storage system; and

writing all dirty blocks in said blockmap file and said inode file to said storage system.

103.1002.12

6. (Cancelled)

4

7. (Currently Amended) A method of maintaining data in a storage system, comprising the steps of:

maintaining a root inode node and inodes for a file system in the storage system, the root inode node pointing directly or indirectly to the inodes, and each inode storing file data, pointing to one or more blocks in the storage system that store file data, or pointing to other inodes, with different levels of indirection possible for the inodes to point to the blocks; and

maintaining an inode map and a block map for the file; ~~and after data in the file system is changed, temporarily storing new data and inodes affected by the new data in memory before writing the new data and inodes affected by the new data to the storage system, using a list of dirty inodes to coordinate writing the new data and inodes affected by the new data to new blocks in the storage system, maintaining old data in old blocks in the storage system, updating the inodes and inode map to reflect the new blocks, and updating the block map, with the block map showing that both the new blocks and the old blocks are in use; whereby a record of changes to the file system is automatically maintained in the storage system~~

wherein when said inodes are maintained in the storage system, each inode only points to blocks in the storage system having a same level of indirection as other blocks pointed to by that inode.

8. (Cancelled)

103.1002.12

9. (Cancelled)

<sup>5</sup>  
10. (Currently Amended) A method as in claim <sup>4</sup>7, further comprising the step of creating a snapshot of the file system by copying the root inode ~~node~~.

<sup>6</sup>  
11. (Previously Presented) A method as in claim <sup>5</sup>10, wherein the block map indicates membership of blocks in one or more snapshots.

<sup>1</sup>  
12. (Previously Presented) A method as in claim <sup>5</sup>10, further comprising the step of deleting a snapshot from the storage system, wherein blocks that are only part of the deleted snapshot are released for re-use by the storage system.

DZ  
<sup>9</sup>  
13. (Currently Amended) A memory storing information including instructions, the instructions executable by a processor to record a plurality of data about a plurality of blocks of data stored in a storage system, the instructions comprising the step of ~~steps of~~ maintaining multiple usage bits for each of said plurality of blocks; ~~blocks~~,

wherein one bit of said multiple usage bits for each of said plurality of blocks indicates a block's membership in an active file system and plural bits of said multiple usage bits for each of said plurality of blocks indicate membership in plural read-only copies of a file system; and

103.1002.12

wherein storing, in said storage system, said multiple usage bits for each of said plurality of blocks are maintained in said active file system.

9

7

14. (Previously Presented) A memory as in claim 13, wherein one or more bits of said multiple usage bits for each of said plurality of blocks further indicate block reusability.

10

15. (Currently Amended) A memory storing information including instructions, the instructions executable by a processor to generate a consistency point for a storage system, the instructions comprising the steps of:

marking a plurality of inodes pointing to a plurality of modified blocks in a file system stored on said storage system as being in a consistency point;

flushing regular files to said storage system;

flushing special files to said storage system;

flushing at least one block of file system information to said storage system; and

queuing dirty inodes after said step of marking and before said step of flushing at least one block of file system information; ~~and requeuing any of said dirty inodes that were not part of said consistency point after said step of flushing at least one block of file system information~~

wherein said step of flushing said special files to said storage system further

comprises the steps of:

pre-flushing an inode for a blockmap file to an inode file:

-6-

103.1002.12

allocating space on said storage system for all dirty blocks in said inode and said blockmap files;

flushing said inode for said blockmap file again;

updating a plurality of entries in said blockmap file wherein each entry of said plurality of entries represents a block in said storage system; and

writing all dirty blocks in said blockmap file and said inode file to said storage system.

16. (Cancelled)

17. (Currently Amended) A memory storing information including instructions, the instructions executable by a processor to maintain data in a storage system, the instructions comprising the steps of:

maintaining a root inode node and inodes for a file system in the storage system, the root inode node pointing directly or indirectly to the inodes, and each inode storing file data, pointing to one or more blocks in the storage system that store file data, or pointing to other inodes, with different levels of indirection possible for the inodes to point to the blocks; and

maintaining an inode map and a block map for the file system; ~~and after data in the file system is changed, temporarily storing new data and inodes affected by the new data in memory before writing the new data and inodes affected by the new data to the storage system, using a list of dirty inodes to coordinate writing the new data and inodes affected by the new data~~

-7-

58

18

103.1002.12

~~to new blocks in the storage system, maintaining old data in old blocks in the storage system,  
updating the inodes and inode map to reflect the new blocks, and updating the block map, with  
the block map showing that both the new blocks and the old blocks are in use; whereby a record  
of changes to the file system is automatically maintained in the storage system~~

wherein when said inodes are maintained in the storage system, each inode only  
points to blocks in the storage system having a same level of indirection as other blocks pointed  
to by that inode.

18. (Cancelled)

19. (Cancelled)

<sup>11</sup>  
<sup>12</sup> 20. (Currently Amended) A memory as in claim <sup>11</sup> 17, wherein the instructions  
further comprise the step of creating a snapshot of the file system by copying the root inode node.

<sup>12</sup>  
<sup>13</sup> 21. (Previously Presented) A memory as in claim <sup>12</sup> 20, wherein the block map  
indicates membership of blocks in one or more snapshots.

<sup>14</sup>  
<sup>12</sup> 22. (Previously Presented) A memory as in claim <sup>12</sup> 20, wherein the instructions  
further comprise the step of deleting a snapshot from the storage system, wherein blocks that are  
only part of the deleted snapshot are released for re-use by the storage system.

103.1002.12

15

~~23~~. (Currently Amended) A system comprising:

a processor;

a storage system; and

a memory storing information including instructions, the instructions executable by the processor to record a plurality of data about a plurality of blocks of data stored in the storage system, the instructions comprising the step of steps of: ~~(a)~~ maintaining multiple usage bits for each of said plurality of blocks; ~~blocks;~~

wherein one bit of said multiple usage bits for each of said plurality of blocks indicates a block's membership in an active file system and plural bits of said multiple usage bits for each of said plurality of blocks indicate membership in plural read-only copies of a file system; and

wherein (b) storing, in said storage system, said multiple usage bits for each of said plurality of blocks are maintained in said active file system.

16

15

~~24~~. (Previously Presented) A system as in claim ~~23~~, wherein one or more bits of said multiple usage bits for each of said plurality of blocks further indicate block reusability.

17

~~25~~. (Currently Amended) A system comprising:

a processor;

a storage system; and

103.1002.12

a memory storing information including instructions, the instructions executable by the processor to generate a consistency point for the storage system, the instructions comprising the steps of: (a) marking a plurality of inodes pointing to a plurality of modified blocks in a file system stored on said storage system as being in a consistency point; (b) flushing regular files to said storage system; (c) flushing special files to said storage system; (d) flushing at least one block of file system information to said storage system; and (e) queuing dirty inodes after said step of marking and before said step of flushing at least one block of file system information; and ~~(f) requeuing any of said dirty inodes that were not part of said consistency point after said step of flushing at least one block of file system information~~

D3 wherein said step of flushing said special files to said storage system further comprises the steps of: (f) pre-flushing an inode for a blockmap file to an inode file; (g) allocating space on said storage system for all dirty blocks in said inode and said blockmap files; (h) flushing said inode for said blockmap file again; (i) updating a plurality of entries in said blockmap file wherein each entry of said plurality of entries represents a block in said storage system; and (j) writing all dirty blocks in said blockmap file and said inode file to said storage system.

26. (Cancelled)

103.1002.12

18

21. (Currently Amended) A system comprising:

a processor;

a storage system; and

a memory storing information including instructions, the instructions executable by the processor to maintain data in the storage system, the instructions comprising the steps of:

(a) maintaining a root inode node and inodes for a file system in the storage system, the root inode node pointing directly or indirectly to the inodes, and each inode storing file data, pointing to one or more blocks in the storage system that store file data, or pointing to other inodes, with different levels of indirection possible for the inodes to point to the blocks; and (b) maintaining an inode map and a block map for the file system; and ~~(c) after data in the file system is changed, temporarily storing new data and inodes affected by the new data in memory before writing the new data and inodes affected by the new data to the storage system, using a list of dirty inodes to coordinate writing the new data and inodes affected by the new data to new blocks in the storage system, maintaining old data in old blocks in the storage system, updating the inodes and inode map to reflect the new blocks, and updating the block map, with the block map showing that both the new blocks and the old blocks are in use; whereby a record of changes to the file system is automatically maintained in the storage system~~

wherein when said inodes are maintained in the storage system, each inode only points to blocks in the storage system having a same level of indirection as other blocks pointed to by that inode.

103.1002.12

28. (Cancelled)

29. (Cancelled)

<sup>19</sup> 30. (Currently Amended) A system as in claim <sup>18</sup> 27, wherein the instructions

further comprise the step of creating a snapshot of the file system by copying the root inode node.

<sup>20</sup> 31. (Previously Presented) A system as in claim <sup>19</sup> 30, wherein the block map

indicates membership of blocks in one or more snapshots.

<sup>21</sup> 32. (Previously Presented) A system as in claim <sup>19</sup> 30, wherein the instructions

further comprise the step of deleting a snapshot from the storage system, wherein blocks that are only part of the deleted snapshot are released for re-use by the storage system.

<sup>22</sup> 33. (Currently Amended) A system for maintaining data in storage means,

comprising the steps of:

means for maintaining a root inode node and inodes for a file system in the storage system, the root inode node pointing to inodes, and each inode storing file data, pointing to one or more blocks in the storage means that store file data, or pointing to other inodes; and

means for maintaining an inode map and a block map for the file system; and

~~means for, after data in the file system is changed, temporarily storing new data and inodes~~

-12-

63

D

103.1002.12

74  
~~affected by the new data in memory before writing the new data and inodes affected by the new data to the storage system, using a list of dirty inodes to coordinate writing the new data and inodes affected by the new data to new blocks in the storage system, maintaining old data in old blocks in the storage system, updating the inodes and inode map to reflect the new blocks, and updating the block map, with the block map showing that both the new blocks and the old blocks are in use; whereby a record of changes to the file system is automatically maintained in the storage means~~

wherein when said inodes are maintained in the storage system, each inode only points to blocks having a same level of indirection as other blocks pointed to by that inode.